Basics of Spatial Filtering and Frequency Domain Filtering

Section 3.5, Section 4.2.3, 4.4

Why perform filtering on images?

Why perform filtering on images?

- Smoothing
- Sharpening
- Denoising

Basics of Spatial Filtering

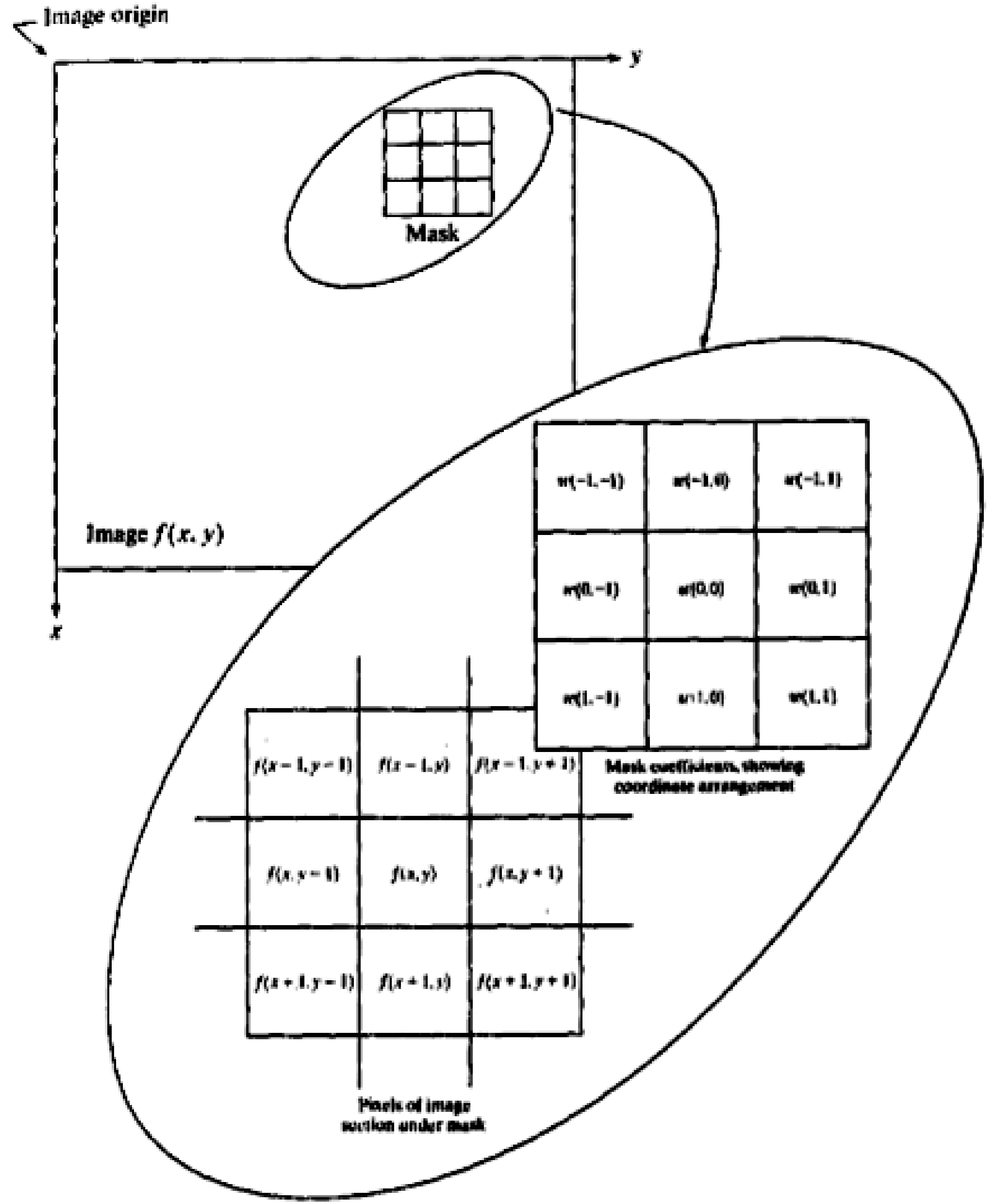
- Subimage, comprising of information from the neighborhood pixels
- Subimages are called filter, mask, kernel, template or window
- Values in filter subimage are called *coefficients,* rather than pixels

Spatial vs Frequency Filtering

Spatial vs Frequency Filtering

- **Spatial:** Operations performed directly on the pixels.
- **Frequency:** Images is tranformed to Frequency domain, and filtering is performed

Mechanics of Spatial Filtering



Mechanics of Spatial Filtering

- Simply move the filter mask from point to point in an image
- At each point the response of the filter is calculated using a predefined relationship

  Response R for linear filtering of 3x3 mask is,

$$R = w(-1,-1)f(x-1, y-1) + w(-1,0)f(x-1, y) + \cdots + w(0,0)f(x,y) + \cdots + w(1,0)f(x+1, y) + w(1,1)f(x+1, y+1),$$

- Concerned with masks of odd sizes

Linear filtering of an image

- General fiter mask of size mxn for an image *f* of size MxN is,

$$g(x, y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s, t) f(x + s, y + t)$$

#convolution

Non Linear filtering of an image

- Sliding the mask past an image is same as in linear filtering
- Filtering is based conditionally on the values of the pixels in the neighborhood
- They do not explicitly use coefficients in the sum of products manner

  Ex: Noise reduction can be achieved effectively with a non-linear filter with a basic function of computing the median gray level value in the neighborhood

  Computation of median is a non-linear operation, as is that of variance

Smoothing Spatial Filters

- Used for blurring and for noise reduction
- Blurring is used in preprocessing
- O/P of a linear smoothing filter is simply the average of the pixels contained in the neighborhood of the filter mask
- Hence called, *Averaging Filters*

Principle of Averaging filters

- By replacing the value of every pixel in an image by the average of gray levels in the neighborhood results in reduced “sharp” transitions in gray levels
- Noise reduction

Disadvantage:

????

Spatial filtering using Averaging filters

$$R = \frac{1}{9}\sum_{i=1}^{9} z_i,$$

$$\frac{1}{9} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \qquad \frac{1}{16} \times \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

A spatial imaging filter with all equal coefficients is called as a Box Filter

Weighted Averaging filter

$$g(x, y) = \frac{\sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s, t) f(x + s, y + t)}{\sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s, t)}$$

Order Statistics Filters (non-linear)

- Non linear spatial filters
- Response is based on the rank of the pixel

  Ex: Median Filter: Replaces the pixel by the median of the graylevels in the neighborhood of that pixel
- Median filters are excellent to tackle certain types of random noise
- Particularly effective in the presence of impulse noise / salt and pepper noise!

Other Order-Statistics filters

- Max filters to yield 100^{th} percentile results
- Min filters to yield 0^{th} percentile results

PS: Median filters provide what percentile results?

Sharpening Spatial filters

- To highlight finer detail in the image
- To enhance detail that has been blurred

  Applications: Ranging from electronic printing and medical imaging to industrial inspection and autonomous guidance in military systems

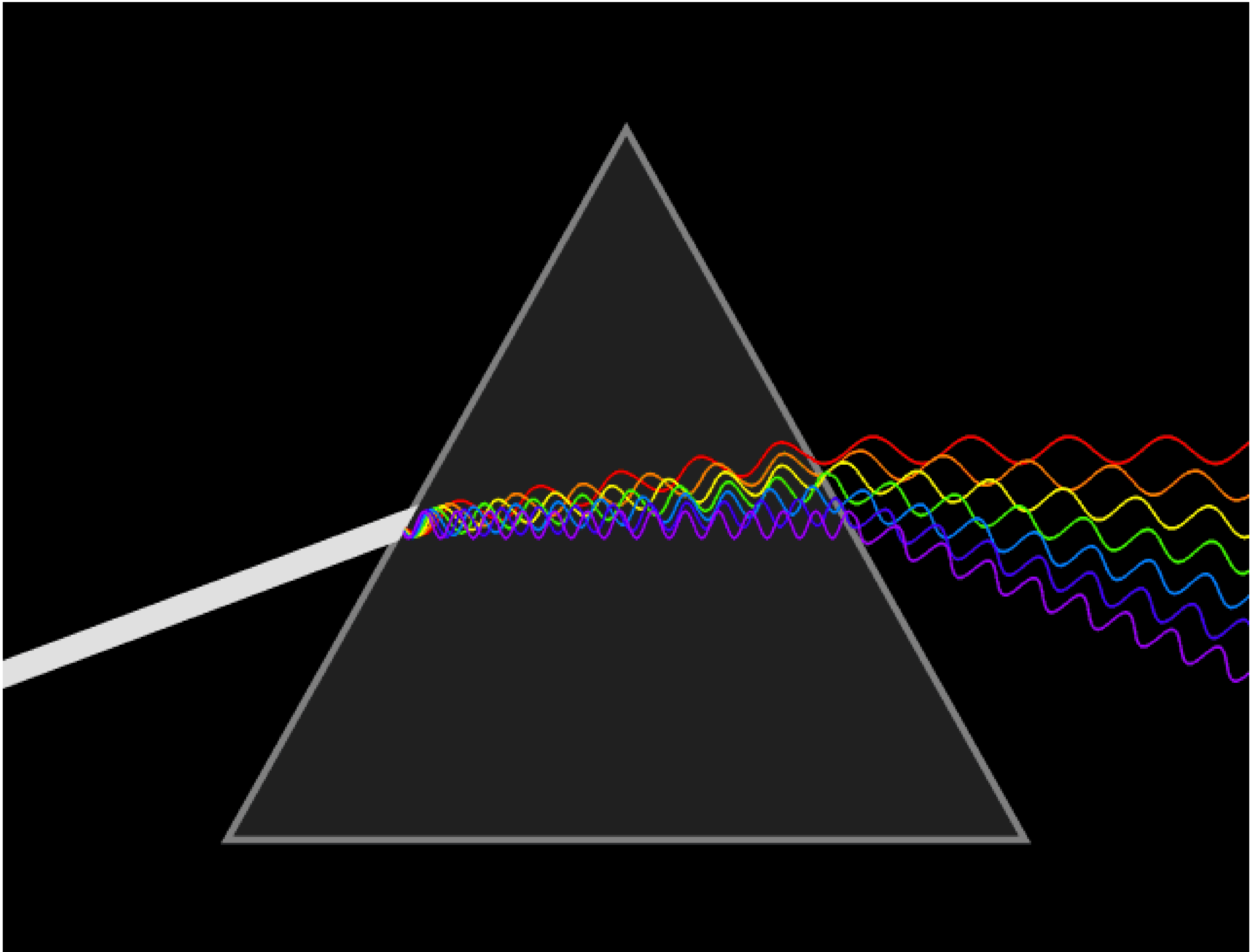
Filtering in Frequency Domain

- Background
- Filtering in Frequency Domain
  - Basic properties of the Frequency Domain
  - Basics of filtering in Frequency Domain
  - Some Basic filters and properties
  - Correspondence between filtering in spatial and frequncy domains

Filtering in Frequency Domain

- Smoothing Frequency Domain filters
- Sharpening frequncy domains
- Homomorphic filtering

Fourier Analysis <-> A mathematical Prism!



2-D DFT Pair

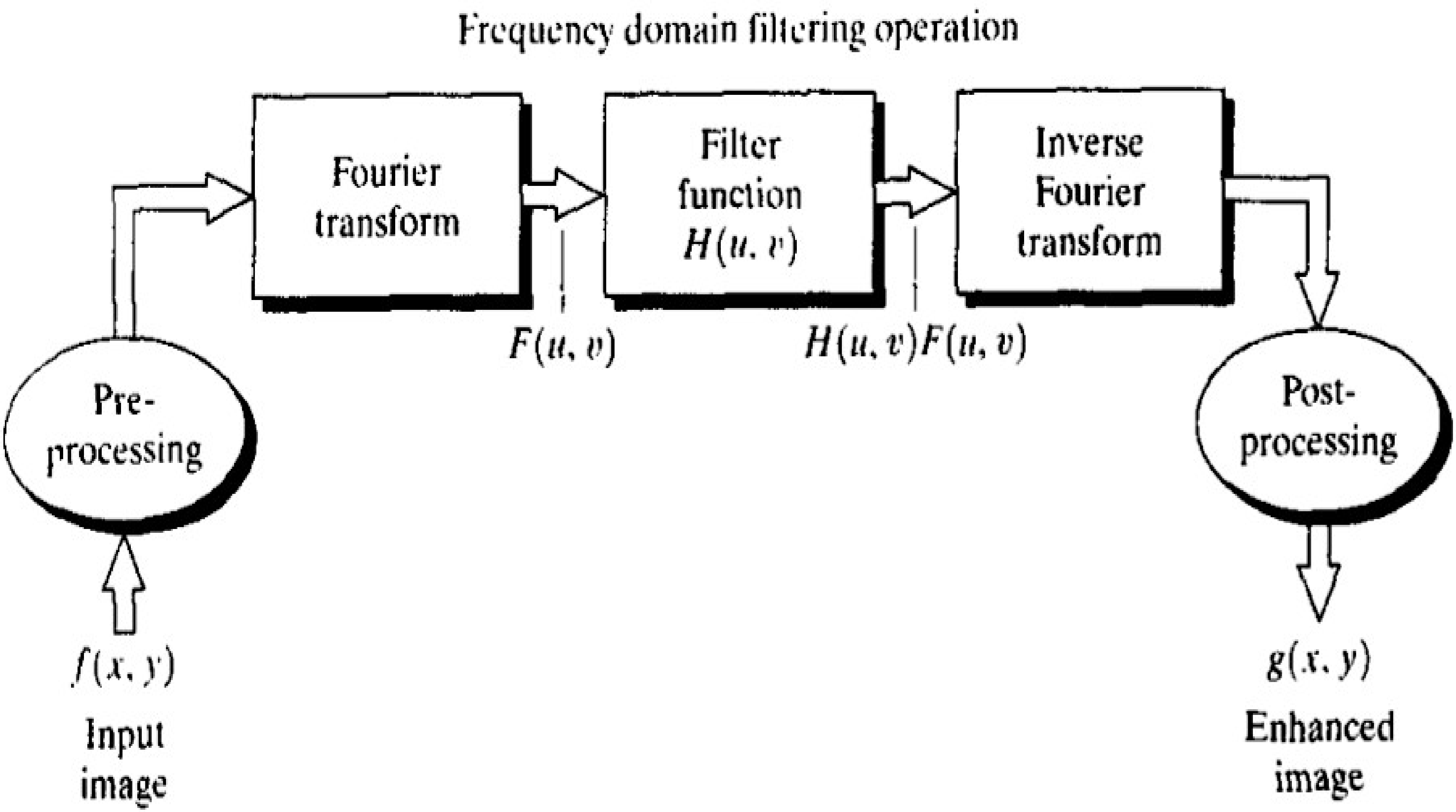
$$F(u,v) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)e^{-j2\pi(ux/M+vy/N)},$$

$$f(x,y) = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F(u,v)e^{j2\pi(ux/M+vy/N)}$$

Properties of Frequency Domain

- With exception of some trivial cases, it is impossible to make direct associations between image and its transforms
- Frequency variations are associated with variations in intensity levels
- Slowest varying frequency component (u=v=0) corresponds to the average gray level of an image

Basic steps for filtering in Frequency Domain



Steps involved in Frequency Domain filtering

1) Multiply the input image by $(-1)^{x+y}$
2) Compute $F(u,v)$, the DFT of the image
3) Multiply $F(u,v)$ by a filter function $H(u,v)$
4) Compute the inverse DFT of the result in (3)
5) Obtain the real part of the result in (4)
6) Multiply the result in (5) by $(-1)^{x+y}$

What are Zero Phase shift Filters?

Filtering and recovering

$$G(u,v)=H(u,v)F(u,v) \quad \text{--Filtering}$$

$$\text{Filtered image} = \mathcal{F}^{-1}[G(u,v)]$$

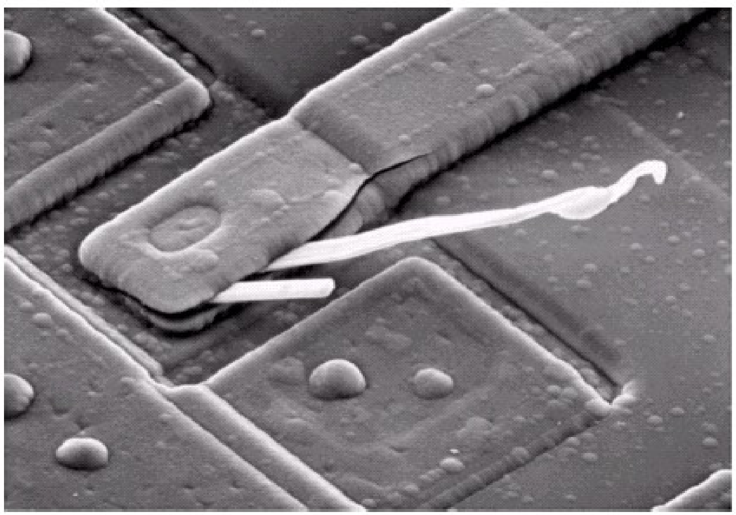
Some basic filters and their properties

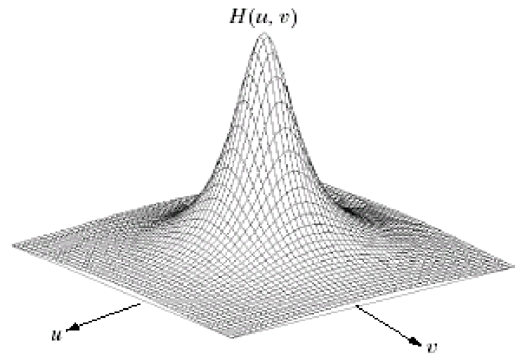
- Notch Filter
- Lowpass Filter
- Highpass filter

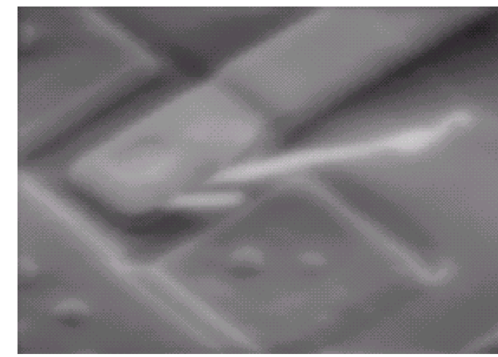
Notch Filter

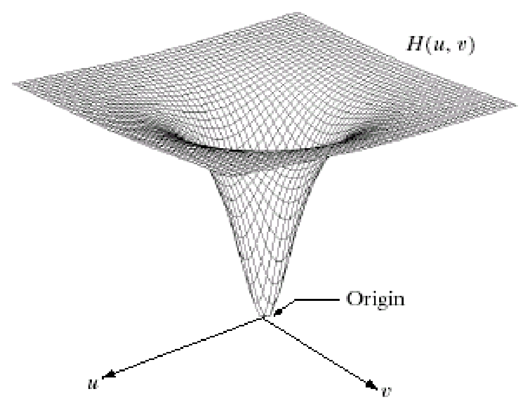
$$H(u,v) = \begin{cases} 0 & \text{if} \quad (u,v) = (M/2, N/2) \\ 1 & \text{otherwise.} \end{cases}$$

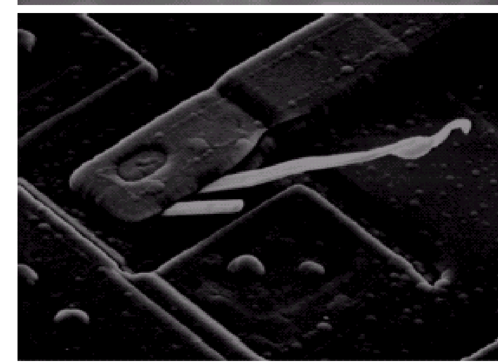
Low Pass Filter











High Pass Filter

Correspondence b/w spatial and frequency domain filtering

Convolution Theorem

$$f(x,y)*h(x,y) \Leftrightarrow F(u,v)H(u,v)$$

Smoothing Frequency Domain Filters

- Edges, sharp transitions (noise) in gray levels contribute to high frequency content in Fourier spectrum
- Smoothing/blurring is achieved by attenuating a range of high frequency components
- Using zero-phase shift filters H(u,v)
- Three types of Low pass filters

i. Ideal

ii. Butterworth

iii. Gaussian

Ideal Low Pass Filters

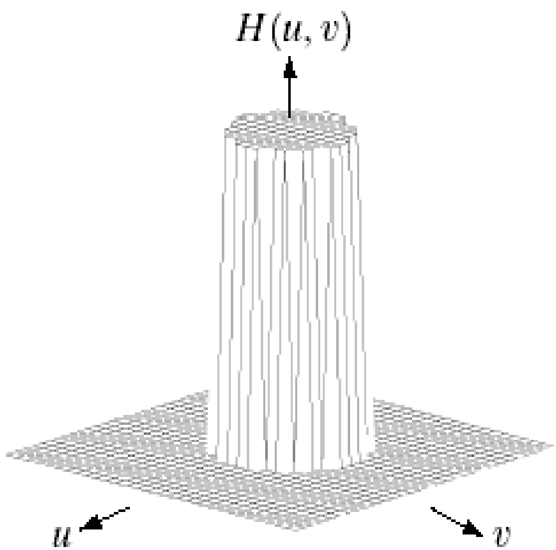
- Cuts off all high freq components of the Fourier transform that are at a distance greater than a specified distance D_0 from the origin of centered transform

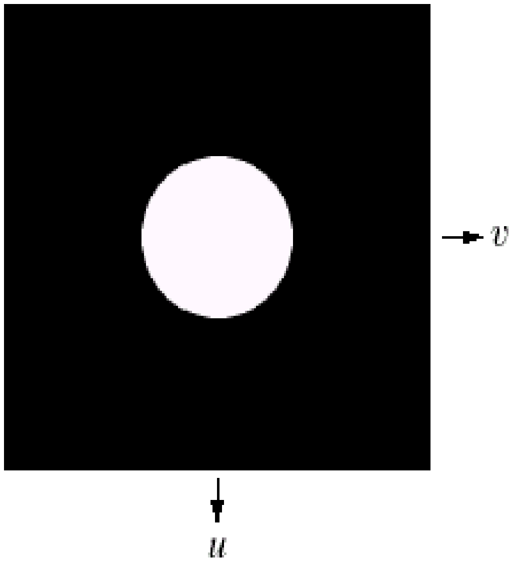
$$H(u,v) = \begin{cases} 1 & \text{if } D(u,v) \le D_0 \\ 0 & \text{if } D(u,v) > D_0 \end{cases}$$

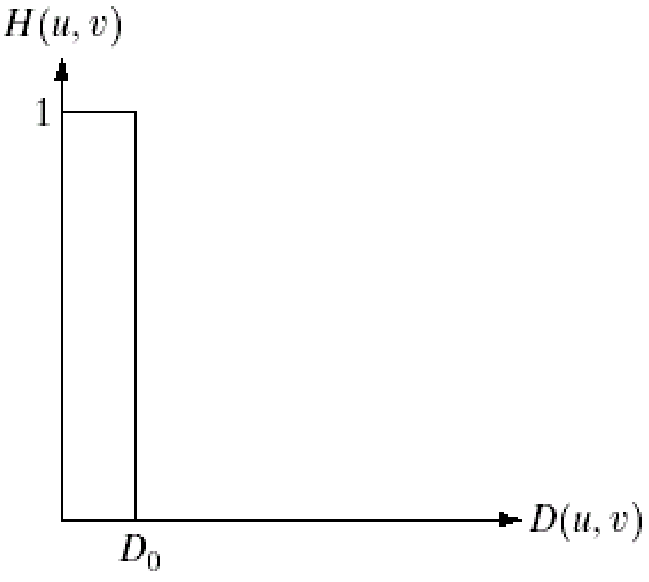
Where D0 is a specified non-negative quantity, and D(u,v) is the distance from the point (u,v) to the center of the frequency triangle

ILPF

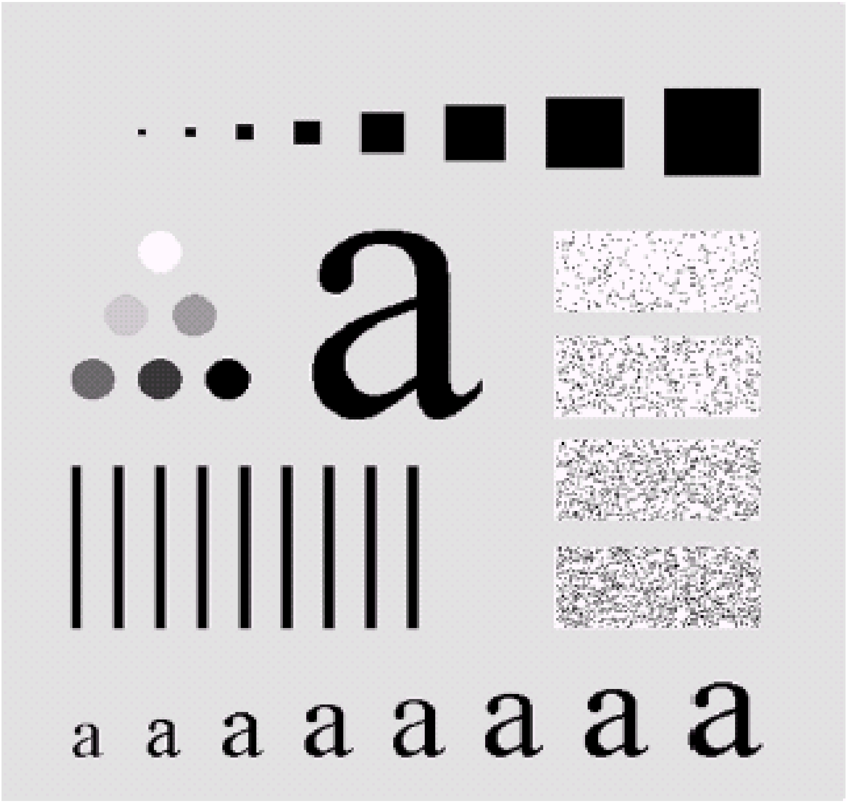
$$D(u,v) = [(u - M/2)^2 + (v - N/2)^2]^{1/2}$$

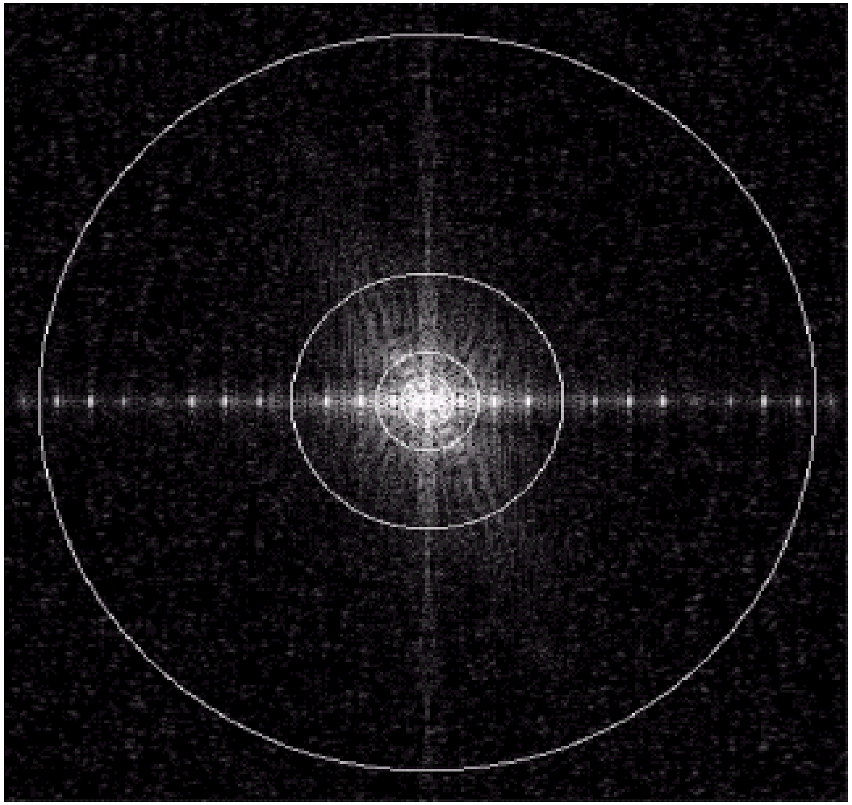




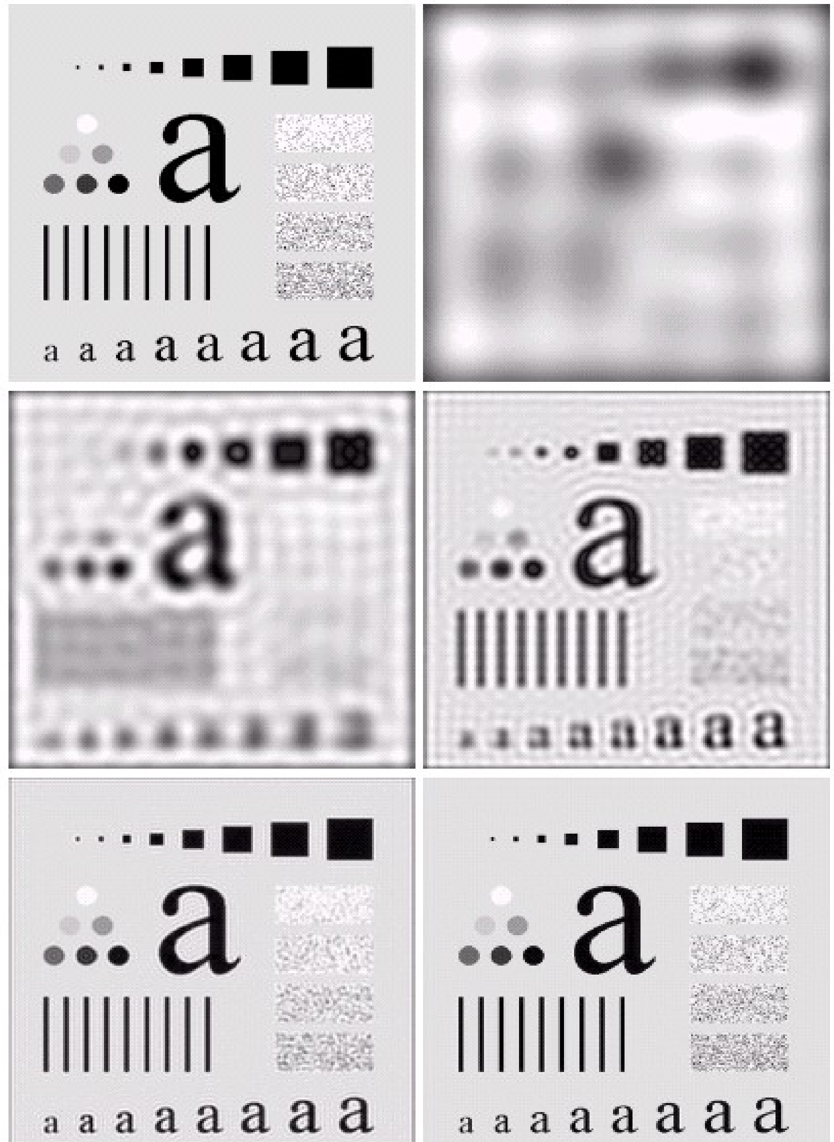


Ideal Low Pass Filtering





Impact of varying the radius of the filter



Original image

Result of filtering with ideal low pass filter of radius 5

Result of filtering with ideal low pass filter of radius 15

Result of filtering with ideal low pass filter of radius 30

Result of filtering with ideal low pass filter of radius 80

Result of filtering with ideal low pass filter of radius 230

Features of ILPF

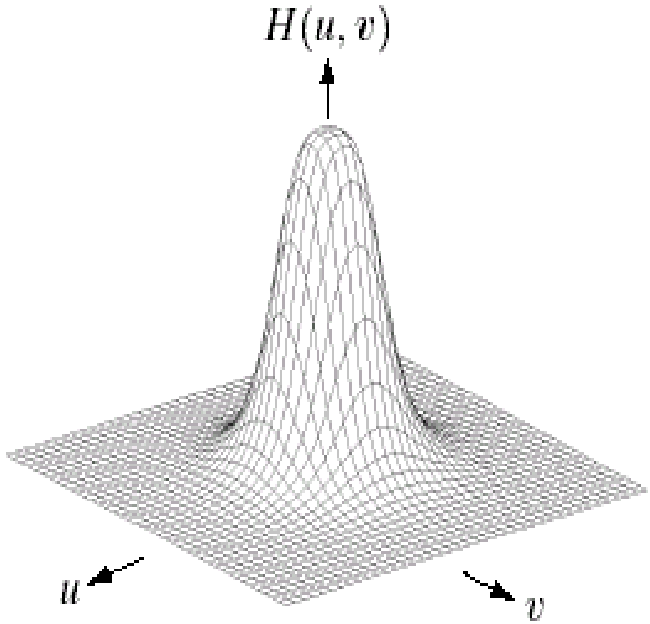
- Dominant component at the origin
- Concentric, circular components about the center component
- Center component is primarily responsible for blurring
- Concentric circles are responsible for ringing charecteristic of ideal filters
- Radius and number of concentric circles are inversely proportional value of cutoff frequency

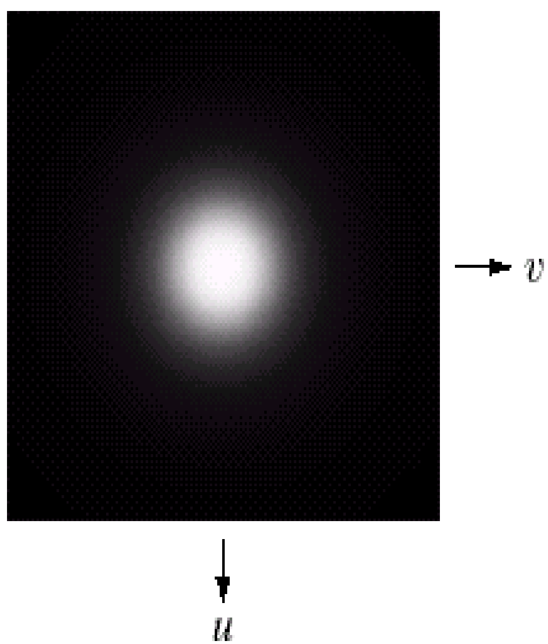
Butterworth Low Pass Filter

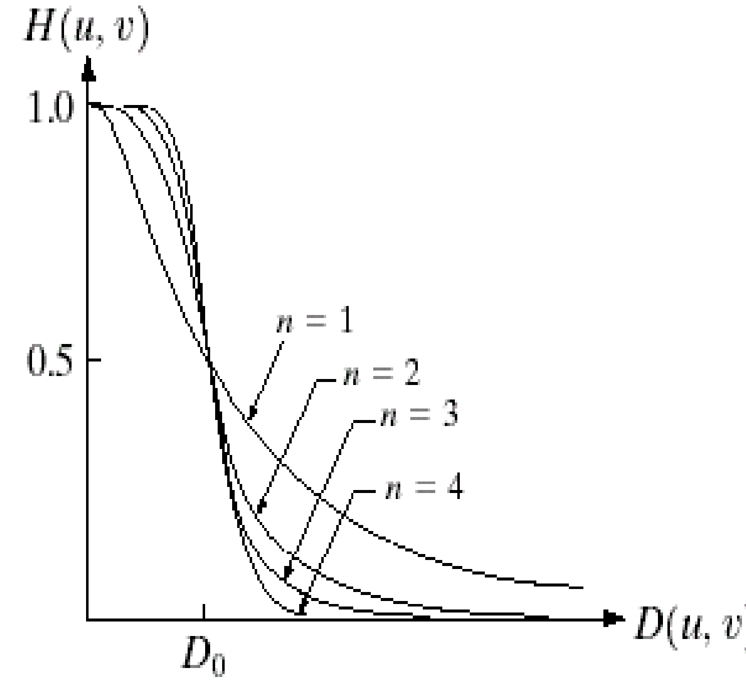
The transfer function of a Butterworth lowpass filter of order n with cutoff frequency at distance D_0 from the origin is defined as:

$$H(u,v) = \frac{1}{1 + [D(u,v) / D_0]^{2n}}$$

Butterworth LPF



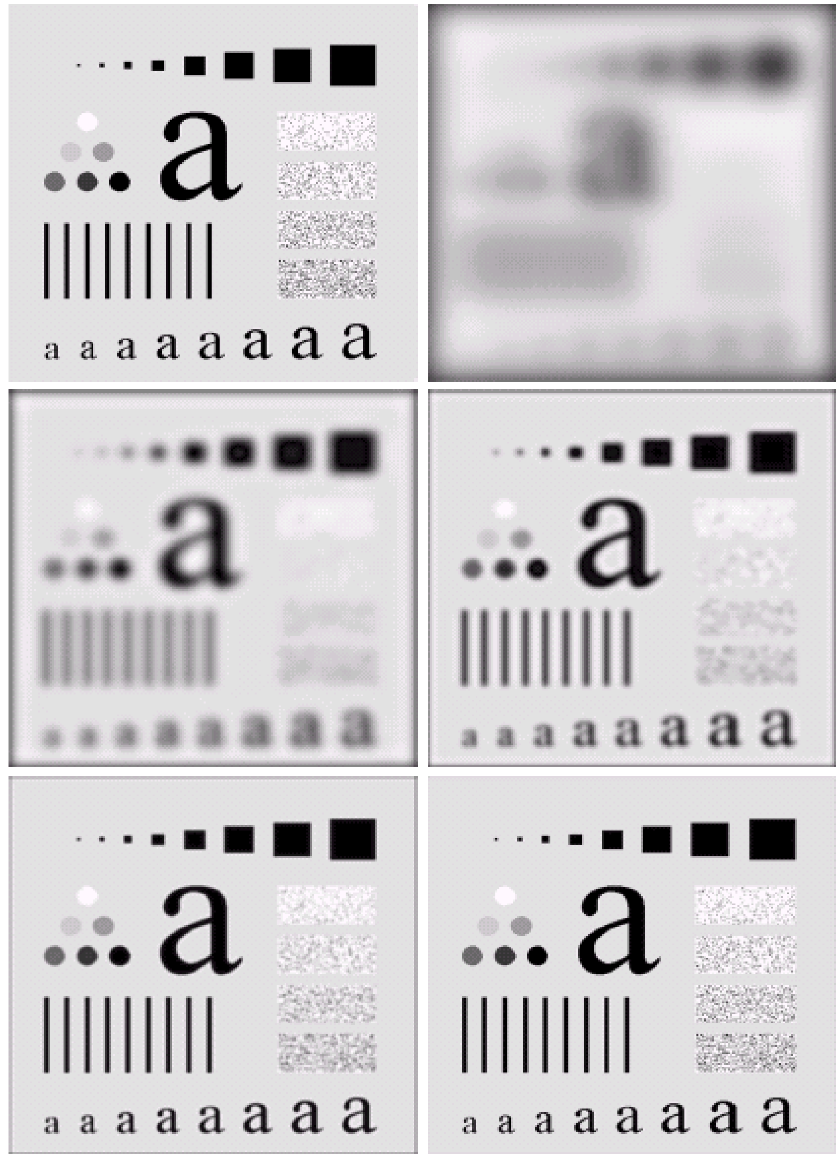




Butterworth LPF

- Does not have a sharp discontinuity establishing a clear cut-off frequency
- Cut off frequency is the locus at points for which H(u,v) is down to a certain fraction of its maximum value . Ex: 50% of its maximum value
-

Butterworth LPF



Original image

Result of filtering with Butterworth filter of order 2 and cutoff radius 5

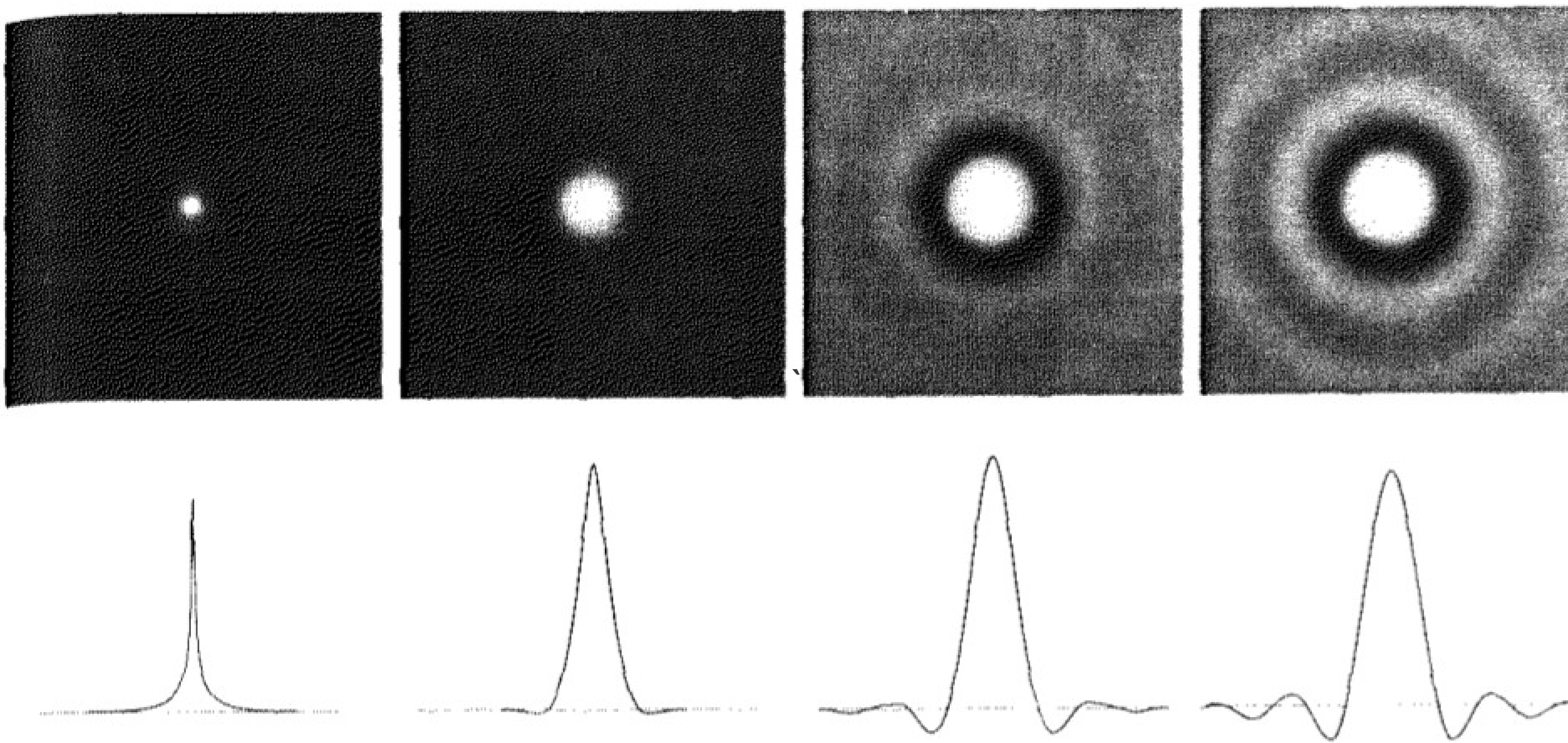
Result of filtering with Butterworth filter of order 2 and cutoff radius 15

Result of filtering with Butterworth filter of order 2 and cutoff radius 30

Result of filtering with Butterworth filter of order 2 and cutoff radius 80

Result of filtering with Butterworth filter of order 2 and cutoff radius 230

Spatial representation of BLPF

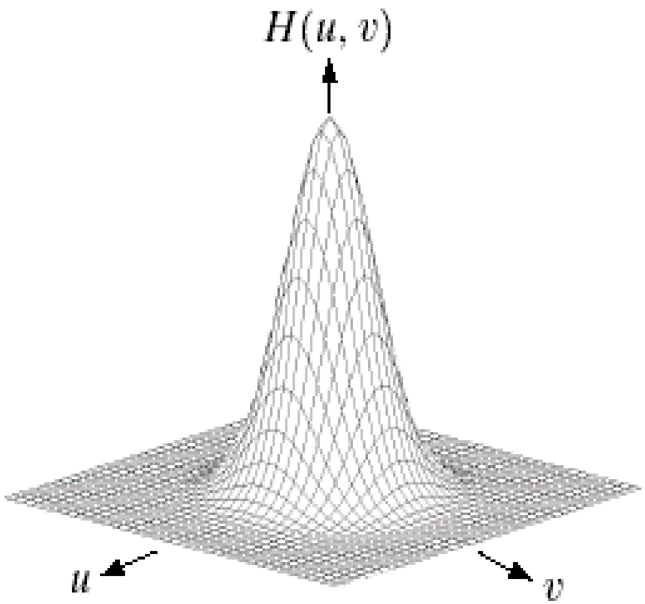


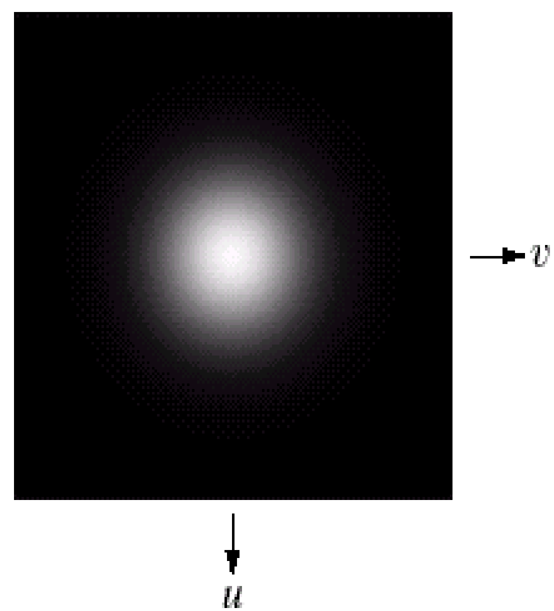
Gaussian LPF

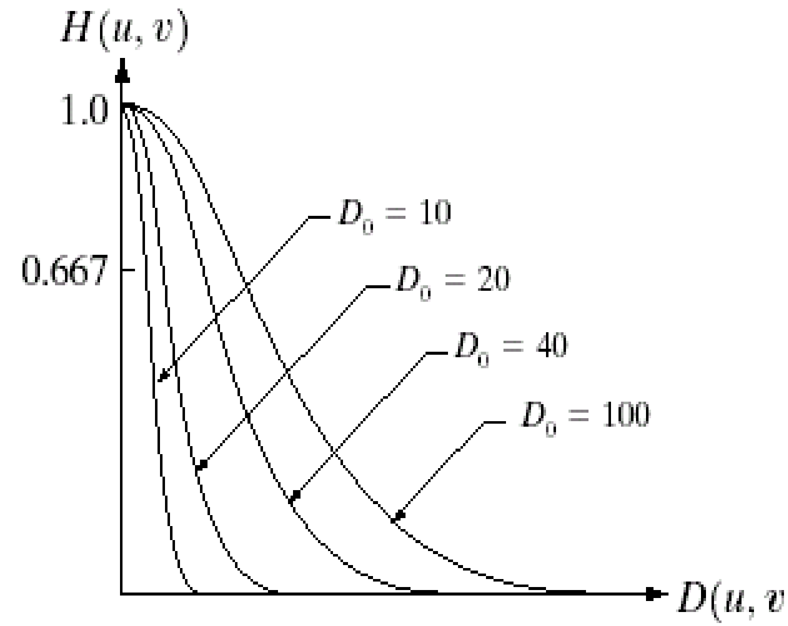
Transfer function of Gaussian LPF

$$H(u,v) = e^{-D^2(u,v)/2D_0{}^2}$$

Gaussian LPF







Sharpening in the Frequency Domain

Edges and fine detail in images are associated with high frequency components

High pass filters – only pass the high frequencies, drop the low ones

High pass frequencies are precisely the reverse of low pass filters, so:

$$H_{hp}(u, v) = 1 - H_{lp}(u, v)$$

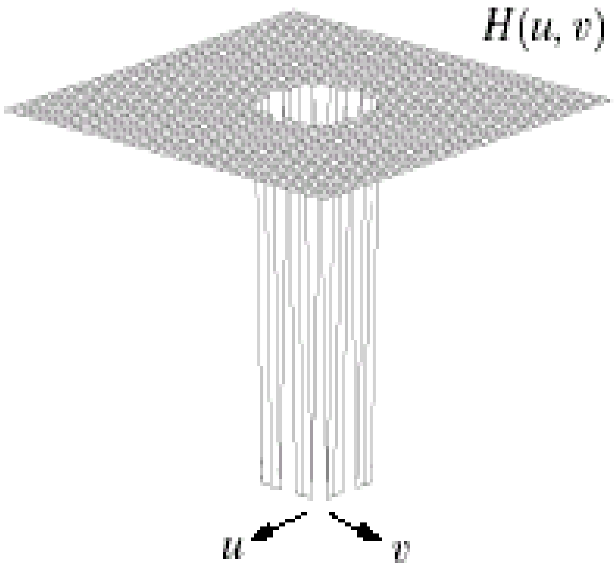
Ideal High Pass Filter

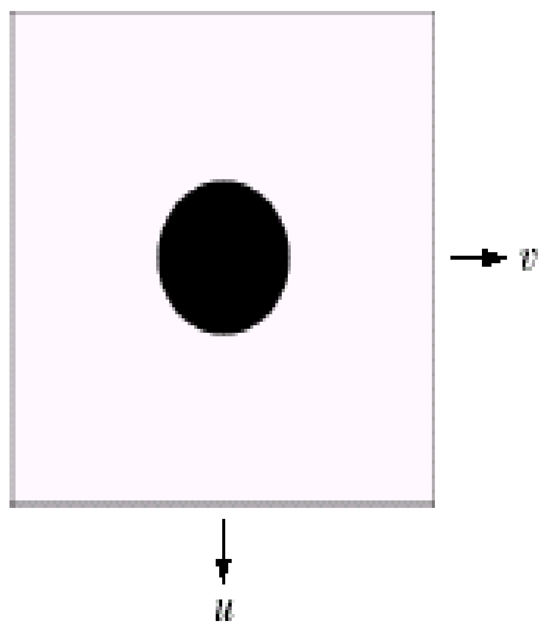
The ideal high pass filter is given as:

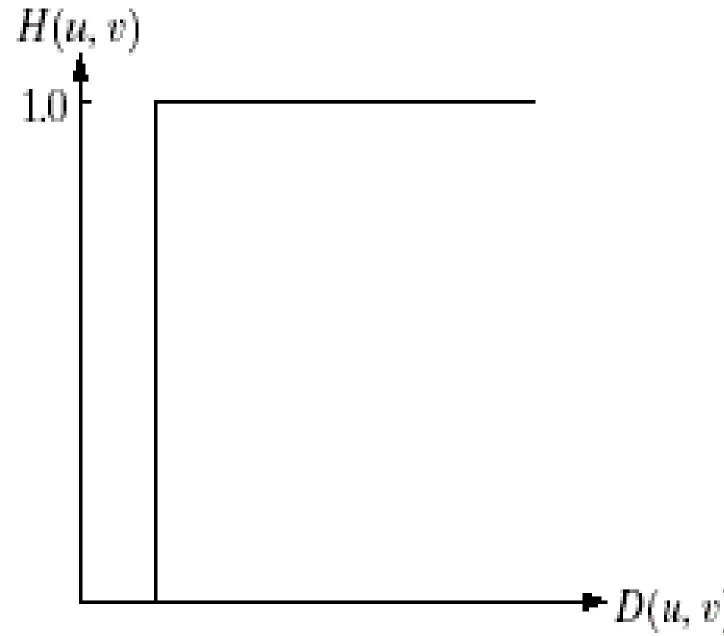
$$H(u,v) = \begin{cases} 0 \text{ if } D(u,v) \le D_0 \\ 1 \text{ if } D(u,v) > D_0 \end{cases}$$

where D_0 is the cut off distance as before

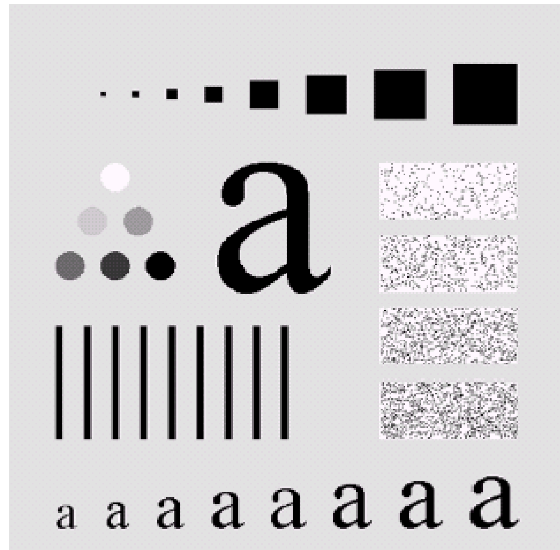
Ideal High Pass Filter

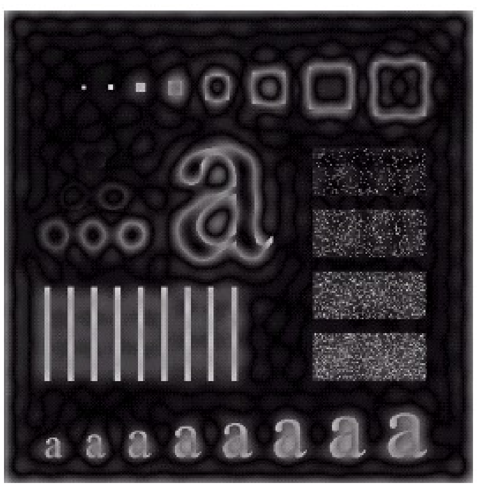




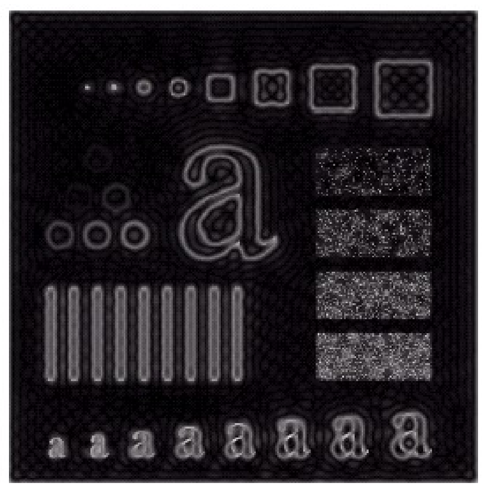


Ideal High Pass Filter

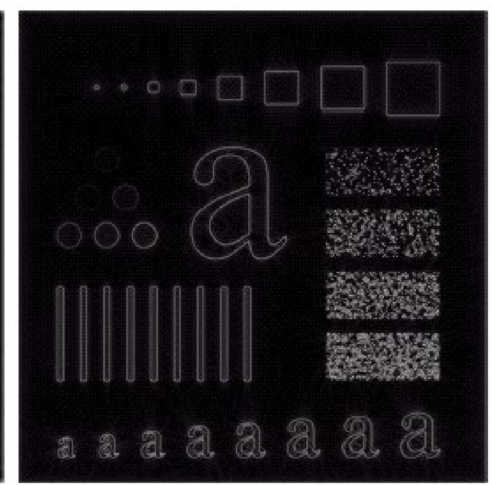




Results of ideal high pass filtering with D_0 = 15



Results of ideal high pass filtering with D_0 = 30



Results of ideal high pass filtering with D_0 = 80

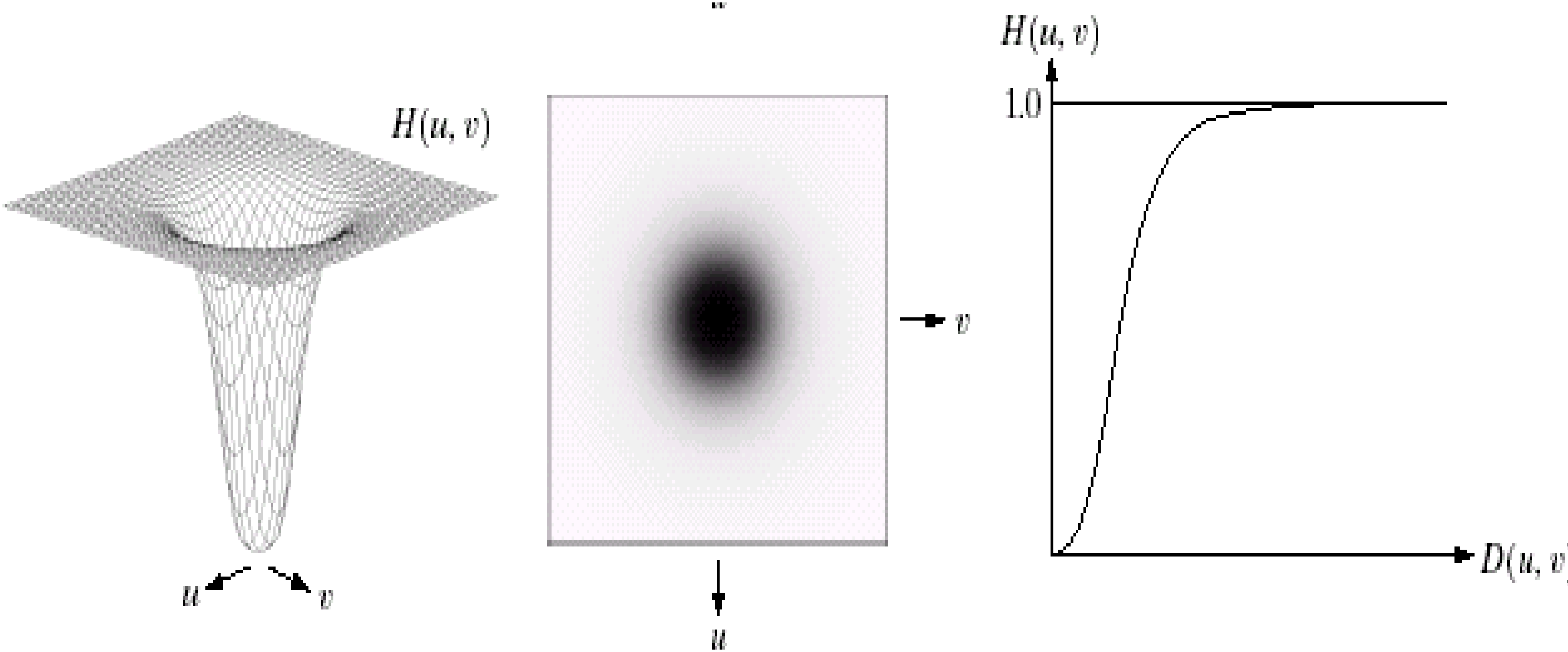
Butterworth High Pass Filters

The Butterworth high pass filter is given as:

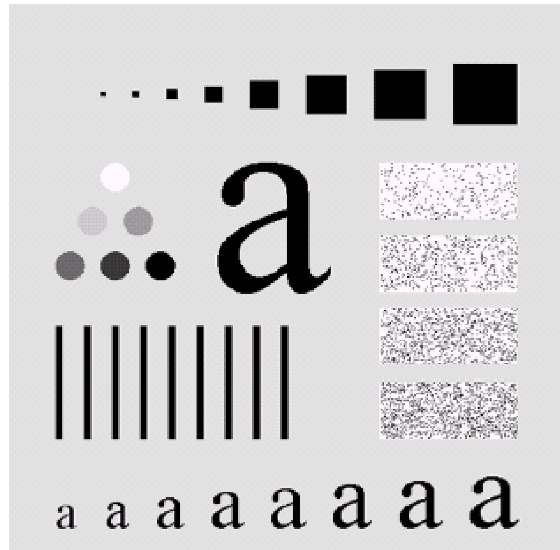
$$H(u,v) = \frac{1}{1 + [D_0 / D(u,v)]^{2n}}$$

where n is the order and D_0 is the cut off distance as before

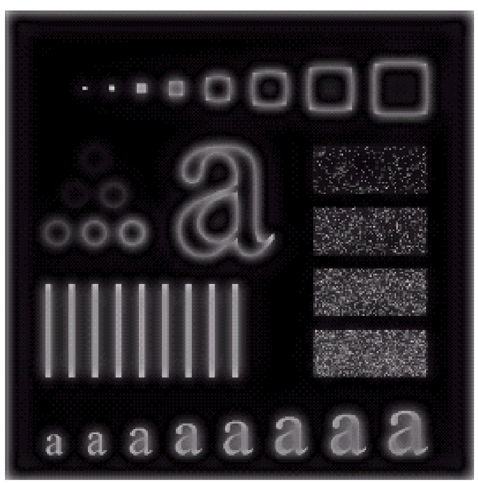
Butterworth High Pass Filters

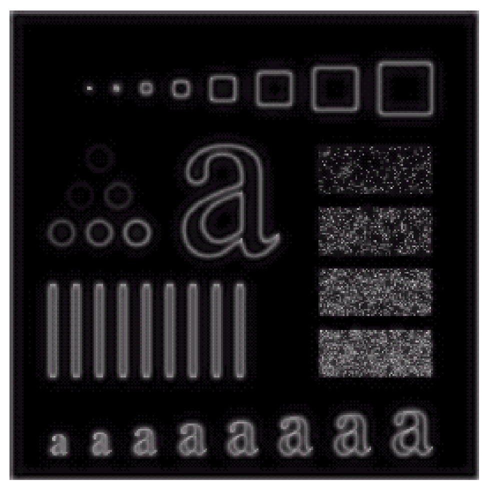


Butterworth High Pass Filters (cont…)

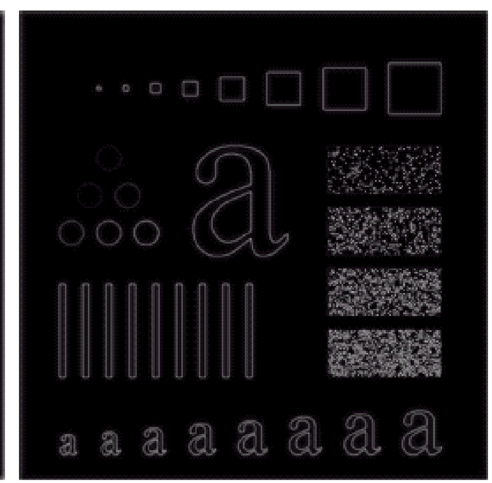


Results of Butterworth high pass filtering of order 2 with D_0 = 15





Results of Butterworth high pass filtering of order 2 with D_0 = 30



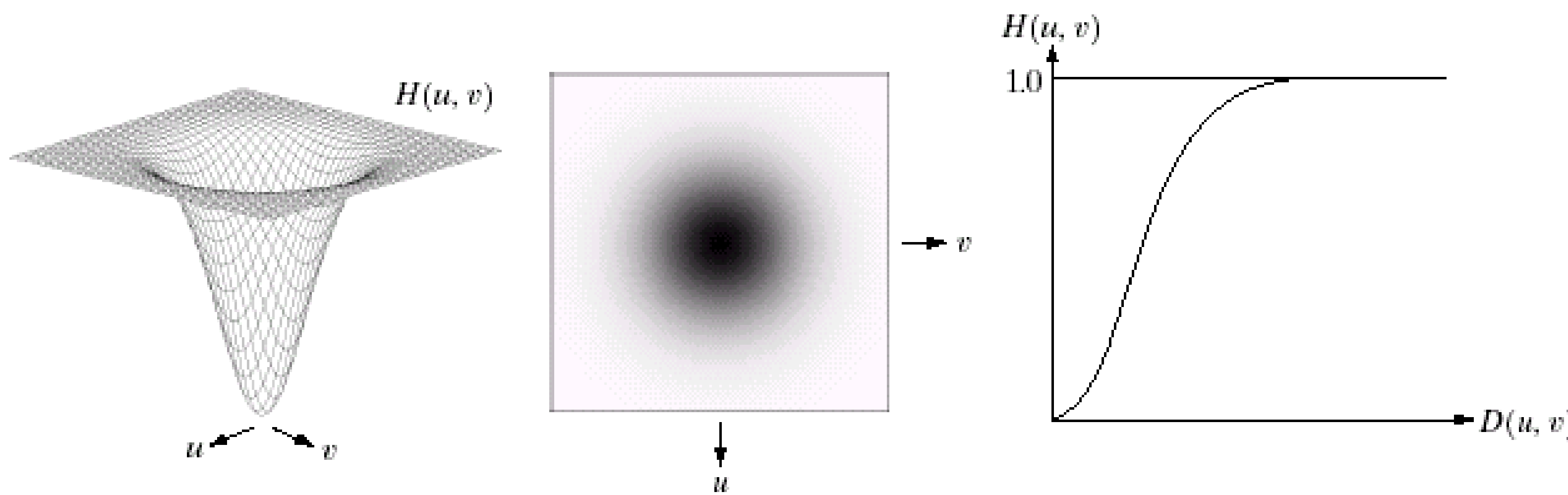
Results of Butterworth high pass filtering of order 2 with D_0 = 80

Gaussian High Pass Filters

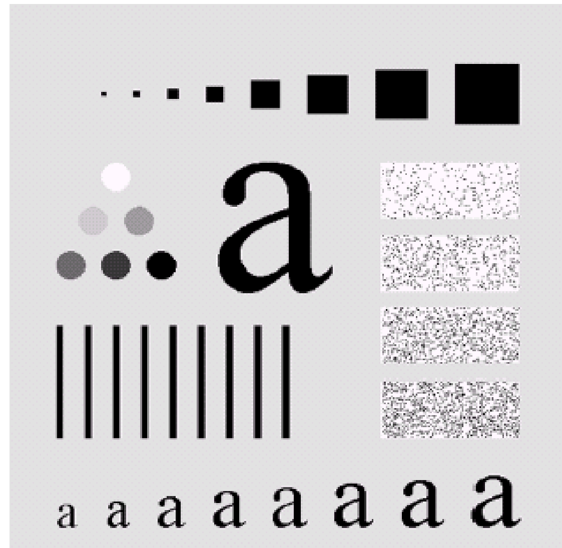
The Gaussian high pass filter is given as:

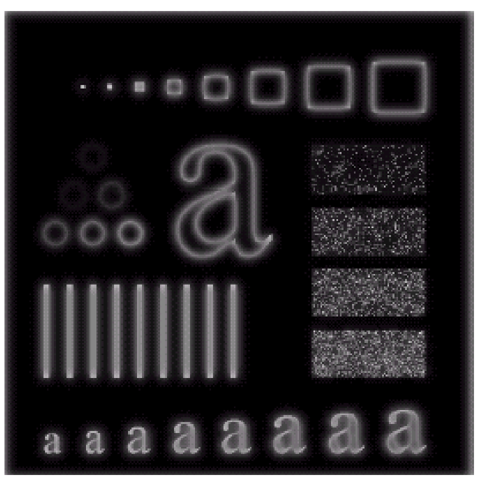
$$H(u,v) = 1 - e^{-D^2(u,v)/2{D_0}^2}$$

where D_0 is the cut off distance as before

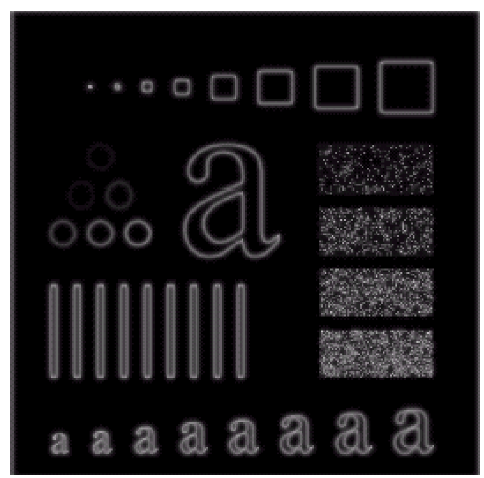


Gaussian High Pass Filters (cont…)





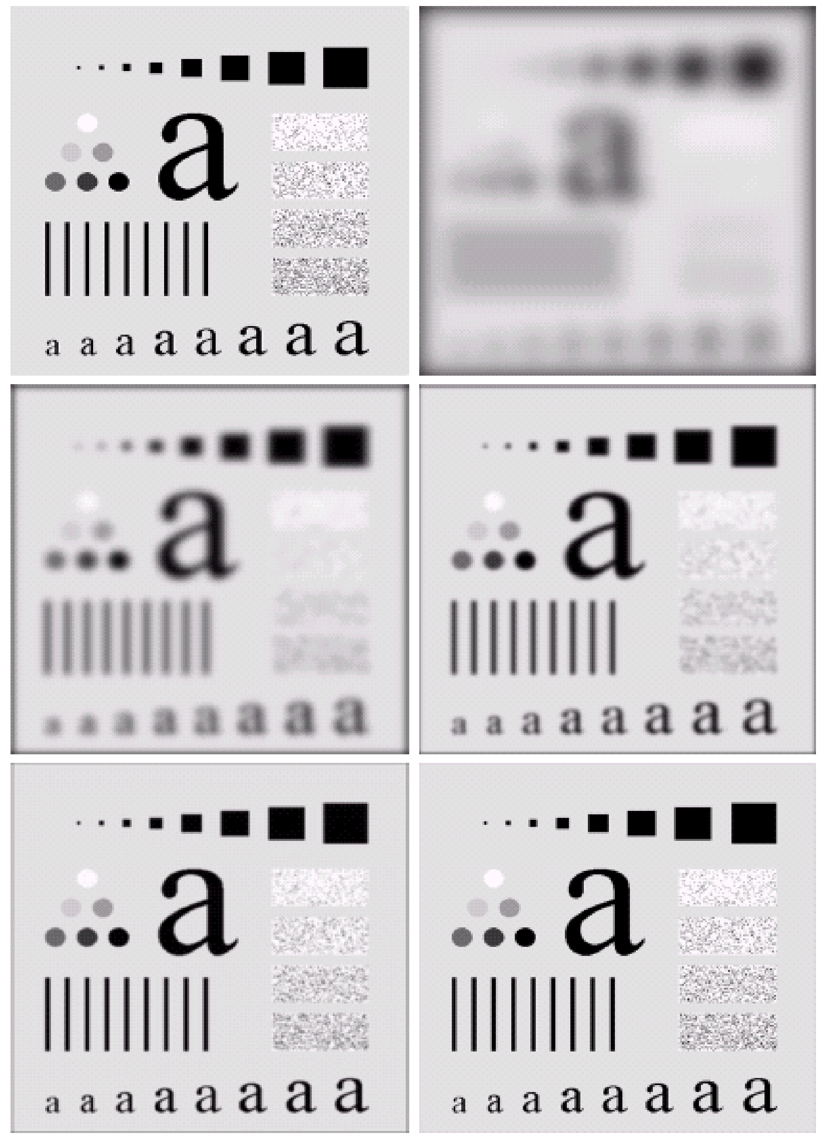
Results of Gaussian high pass filtering with $D_0 = 15$



Results of Gaussian high pass filtering with $D_0 = 30$

Results of Gaussian high pass filtering with $D_0 = 80$

Gaussian LPF



Original image

Result of filtering with Gaussian filter with cutoff radius 5

Result of filtering with Gaussian filter with cutoff radius 15

Result of filtering with Gaussian filter with cutoff radius 30

Result of filtering with Gaussian filter with cutoff radius 85

Result of filtering with Gaussian filter with cutoff radius 230

Homomorphic Filtering